AD-A120 248 DUKE UNIV DURHAM NC DEPT OF CHEMISTRY F/G 12/1
SEARCH ALGORITHMS AND THEIR IMPLEMENTATION.(U)
AUG 82 D W LOVELAND AFOSR-81-0221
UNCLASSIFIED AFOSR-TR-82-0878 NL

1 of 1

END
DATE
FILMED
11 82
DTIC

(4)

ANNUAL REPORT, AFOSR-81-0221

'SEARCH ALGORITHMS AND THEIR IMPLEMENTATION'

August 1982 (1 Jul 81-30 Jun 82)

D.W. Loveland
Computer Science Department
Duke University
Durham NC 27706

ABSTRACT

Research that has resulted in completed papers involved (1) optimality of search procedures (decision trees) in binary testing; (2) a study of signature table representation for evaluation functions and methods for dynamically improving function accuracy; (3) pruning minimax trees that have been adapted to incorporate moves determined by chance; and (4) the search problem in automated program construction. Preliminary results have been obtained in research on (a) optimizing limited resource use to guide otherwise random search; (b) studying search strategies in two-person games when information is partly concealed; and (c) limiting search in debugging rule sets in one type of expert knowledge system. Other investigations are in progress.

ANNUAL REPORT

Search Algorithms and Their Implementation
(AFOSR 81-0221)

June 30, 1981 - June 29, 1982

Overview

Support from the Air Force Office of Sponsored Research is now one year old and we feel that it has been generally a very successful year. A paper accepted, a paper submitted, a book chapter being published, and a paper to be presented at the AAAI Conference this August (1982) indicates that results are already flowing to the outside world. Other investigations underway are obtaining results, and research papers will be written when appropriate. A seminar in search methodology was run for two semesters with considerable success, including enticing a researcher (Ballard) into the area whose work has already achieved recognition (presentation at the AAAI conference). Results have emerged in several areas of search not anticipated.

A few disappointments occurred also. One investigator (Bowyer) found himself overloaded and did not participate as much as expected and has agreed to be replaced by Ballard. Work in minimum variance search has not progressed as quickly as hoped. (The idea seems to be a good one intuitively; at least two other researchers around the country are now considering it.)

In general, our plusses well outweigh our minuses.

Research objectives

The overall research objective is to further our knowledge regarding the methodology and uses of search. We are pursuing a number of different investigations in this regard, and research objectives will be given for each separately. We limit ourselves here to a brief overview.

Search is a very important component of problem solving, particularly to "artificial intelligence" (AI) tasks where solutions to problems cannot be expressed in closed form or even in problem specific recurrence relations. However, except for certain tasks within operations research (e.g. linear programming) search techniques and tasks involving them have not been methodically investigated. Some successful work has been done within the AI field, notably regarding the A^*-search procedure and regarding minimax search pruning (alpha-beta pruning). Much more needs to be done.

Specifically, we are interested in alternate search methods to minimax search; search methods for settings other than deterministic, two-person zero-sum games with perfect information; searching with limited resources; methods for optimizing search trees in a specific setting (testing); and limiting search cost in debugging rule sets in expert knowledge systems. This last interest is leading us to a direct interest in expert knowledge systems, an important area of research in its own right.

More specific research objectives are given within the research project summaries that follow.

Research status

The status report is divided into projects with the personnel involved named in parentheses. Graduate students' names appear before their advisor's names when the role of the advisor has been limited to problem formulation and/or research guidance.

Binary testing (Loveland).

Here the objects of study are searches, called decision trees, for one distinguished object among many objects. Given are the number of objects, the a priori likelihood that each is the distinguished object, a set of tests, each test identified with a subset of the objects (one considers the test responding "positive" if the distinguished object is in the subset), and a cost associated with each test. The general research objective here is to find good algorithms for finding efficient search procedures for isolating the distinguished object. This is a much studied field by applied statisticians, in particular within botany, because of the need to run classification tests. Computer scientists have done work here because of interest in fault analysis.

Results were obtained extending work by Garey and Graham beyond the case of unit-cost tests and equal a priori probabilities. They showed that the most popular algorithm used in this endeavor could be fairly bad in some situations. Although the bad situations known to date are perverse, hardly anyone previously believed the algorithm would be at all bad in this situation. We studied the arbitrary a priori probability case where the standard algorithm can fail .UL very badly in .UL natural ways, and show that if a certain condition holds on the set of tests then the standard test is almost as good relative to optimal as for the special case when the a priori probabilities are equal (the Garey-Graham case).

A comment on research objectives here. We have worked off and on over several years seeking better (fast) algorithms than the standard one. This has been unproductive. (The problem is known to be hard.) As just reported, we have had success in better understanding the standard algorithm and some more useful work remains. However, at present we are pursuing a potentially very interesting aspect of this general problem, namely, introducing treatments with the tests because people do not always fully isolate a faulty object before treatment. Some initial progress has been made here.

Debugging rule sets (Loveland, Valtorta).

Here we push our interest and knowledge of search into the area of expert knowledge systems. The research objective is to develop tools to aid the designers of expert knowledge systems develop an inference system and knowledge base that does what they want it to do. In particular, at this time we are emphasizing simpler expert systems that can be implemented using rule sets. Several well-known systems are of this type (e.g. MYCIN) and even languages have been developed to aid in devising such systems (e.g. RITA, ROSIE, OPS5). A very important property of most of these expert systems is flexibility, the ability to easily update the expertise by standardized modification techniques. Use of rules of the form "IF...THEN..." to describe the (higher level) relationships makes such modification easier. Our present concern is to help discover conflicts in the rule set when the knowledge is designed or modified.

We have designed some algorithms to advise the user when certain properties of the rule base are being violated. Many expert systems are classifiers; they are diagnosing a situation by processing an input vector (interactively given) and eventually assigning it a class (perhaps to take action on the class; MYCIN fits this description). The algorithms we have developed inform the user if an added rule now leads to ambiguous classification, for example. This is non-trivial because it asks if there exists an input vector leading to ambiguous classification. This naively requires checking all inputs, an overwhelming task. Other related questions are also answerable. We originally thought that time consuming theorem-proving techniques might be needed but have found "quick" graph theoretic algorithms for this particular debugging.

Some unanswered questions remain to be resolved before a paper is written on this work.

Graduate course advisor expert system (Valtorta, Loveland)

A rule-based expert system is being written primarily as a learning experience for us all. We feel that it is important to actually design and build a system so that our work on debugging expert systems will be aimed towards real problems we have encountered. The subject chosen was selected for its simplicity and our "expertise" in it.

Program synthesis: dealing with search (Biermann).

A chapter "Dealing with Search" was written for a book on program synthesis. The chapter builds on previous work by Biermann in program synthesis (not supported by this grant) which this chapter reviews and an important point is made regarding the search problem for program synthesis using these reviewed techniques. In essence, the point regarding search is as follows. The traditional approach to controlling search within program synthesis has been to use heuristics to trim a very large search space. The proposal is that a better approach may be to modularize the search space so one can in effect "divide and conquer". Find subdomains where programs can be synthesized easily and then decompose tasks into subdomains when possible. It is shown how to handle two particular subdomains.

Signature tables (Biermann _et_ _al_.)

Work started earlier was concluded and a paper submitted pertaining to signature tables and learning. An extensive motivational treatment regarding this subarea appeared in the original AFOSR proposal (under "State evaluation schemes" and so we give a very brief statement here.

In solving problems using search, a critical aspect is the evaluation function that measures the worth of the system at the end of the lookahead along each of the paths followed (the leaves of the search tree). Certain properties of the system are selected as pertinent and used in the evaluation. The proper choice of which properties is usually very difficult so methods have been proposed for dynamic adjustment of the property vector. This is referred to as "learning". The appropriate representation of the function schema for mapping the property vector to a real number is crucial to the success of realizing useful evaluation functions.

Search in the presence of uncertainty (Wagner)

We are currently investigating a generalization of the "search" paradigms with the following characteristics:

a. Two "sides" compete for some goal;
b. Each side makes decisions alternately;
c. Some information about the starting position is concealed from one or both sides;
d. Moves simultaneously expose some concealed knowledge, and result in gain or loss (more or less additively).

Note that randomness or uncertainty exists initially, but is not injected into the situation by anything inherent in the rules of competition, such as required throws of dice.

This problem setting seems complex, but one with fairly wide applications. In addition, in a simplified, "game" setting, it reveals unexpected properties, such as a game-tree search which does not appear to fit the min-max theoretical model of, say, chess, at all. Finally, we have had encouraging results in developing computer algorithms which can rapidly produce sets of "strategies", which necessarily include all possibly-optimal strategies, independent of any assumed knowledge of the probability distribution of the initial (partly random) situation.

There are several possible applications for this methodology, we believe.

In medical diagnosis,we can seek optimal treatment sequences. This apprroach differs from the binary testing work of Loveland (above) in that patient reaction to the prior sequence of treatments can be considered in choosing the next move (at an increase in computation time).

In military operations planning, we can seek optimal tactical strategies which considers the possible reactions of the enemy commander to a sequence of our moves.

Note that both of the above scenarios involve the exposure of previously concealed information during "play".

Our concrete results concern the game of Bridge, in an attempt to develop algorithms in a simple setting. We note that card play at Bridge forms a perfect example of the problems we are addressing. Furthermore, the Bridge card play problem does not seem to have been as carefully investigated as has Chess, or Backgammon, by Artificial Intelligence workers. We suspect that their concentration on "perfect knowledge", or "pure random" games has perhaps overlooked some methods used by humans in more real-world "games". Indeed, the methods we are developing could be valuable as heuristics in "perfect knowledge" games, which become so complex that, in fact, no one, human or machine, truly has the "perfect knowledge" of the position's value that mathematically is present.

Bridge card play by "declarer" takes the form of a four player, but two-sided game. Declarer (South) sees both his own hand and his partner's, and makes all decisions during play. The "other" side, conventionally East and West, constitutes the "defense". Declarer cannot tell which of the two defenders holds each card in the deck not held by North and South. Furthermore, declarer has at his command various kinds of card-plays, each of which works better in some distributions of the cards between East and West than in others.

To make this problem more tractable, we have simplified the actual situation considerably;

a. We consider only No-Trump bridge play;
b. We assume that the defenders have "perfect knowledge" of the placement of every card around the table;

c. We assume that the defenders have "perfect knowledge" of declarer's future strategy;

d. We reduce the number of "suits" in the deck;

e. We may vary the number of cards in each suit.

Our purpose is to cause a computer to rediscover the principles laid out in an early book [Watson] on card play strategy, from essentially <u>no</u> knowledge specific to Bridge. That is, we do not intend to build in heuristics peculiar to Bridge; instead, we hope to cause the computer to winnow out the available strategies, and return those that are "good" in some sense, based strictly on:

(1) the "goal" - which we take to mean "maximize the number of tricks won by North-South"; and

(2) the rules of Bridge, defining "trick", and requiring that players follow suit, if possible.

The research status as of June 1982 is:

1. We have produced a one-suit player which "solves" for the set of optimal strategies in less than 3 minutes, for problems with up to 16 cards in the single suit. Analysis shows that, in fact, the strategies the program discovers satisfy our requirements.

   Regardless of the probability distribution of the original distribution of cards between East and West, some strategy in the set is optimal.

Interestingly, relatively few strategies tend to be returned: fewer than 10, usually.

2. We have produced a two-suit player, with similar properties, for up to 20 cards in the two suits combined (which need not be equal in length). Our program becomes slower, and uses much address space in this setting, however. A Master's thesis, by Robert Fabrizio, has been written describing these results in detail.

Parallelism in search procedures (Bowyer, Wagner).

A literature search for current work was undertaken and models for parallelism explored. The research objective is to understand how search procedures can benefit by execution in a parallel computation environment. A model for parallel computation (the Boolean Vector Machine) has been defined and the architecture specified to the extent that a prototype is contemplated (Wagner). This work is proceeding under separate funding. It seems premature to attempt specific analysis of search strategies on this machine given the intensity of effort needed to complete the basic design and realization of the BVM. This problem is of real interest and is expected to be addressed in the future.

Maximizing payoff with limited resource expenditure (Mutchler, Biermann, Loveland)

As part of our general study of search methodology, we have addressed the question of how effective can search be under limited resources. The general format of the problem is easily stated: person X undertakes a task involving many decision points. At any decision point he can spend some resources to determine which seems the better direction to pursue, but his resources (or capabilities) are limited and at some decision points he may need to take a random guess. How should he use his resources to maximize his payoff?

For our first approach to this problem we simplify the payoff to 0 or 1 and allow only one inquiry. Thus the person in effect begins at the root of a tree and at each node decides to go left or right. Usually it is a random choice, equally likely each way. At some point the person can spend his resource, he can pick the node he is on or a node "ahead" of him and "see" both branches of the node. By "see" we mean that he learns for each side (only) how many 0's and 1's there are on that side. He is presumed to take the next move to the side with the most 1's.

For this problem we now know quite a bit about how much the person gains on the average by having the ability to get information at some point to improve his performance. For example, we can show that he is best off waiting until the last move to ask for information, and that indeed the earlier he asks the less improvement over random performance he realizes. This is true no matter how many 1's are present except for 0,1, n-1, or n ones when n is the number of leaves to the tree. (For these cases the gain is the same anywhere; note that there is a gain over random walk for 1 one or n-1 ones but he can ask anywhere equally effectively.) Actually, the results just stated hold for other than 0 and 1 payoffs as well.

We next intend to study the performance improvement when multiple questions can be asked. Some of the combinatorics may become unmanageable but we hope several natural questions are answerable.

Probability backup search (Truscott, Loveland, Biermann)

An attempt was made to understand an alternate search methodology to minimax. The concern is how one utilizes information gathered by searching to make the best ("correct") decision, as to the next move. Probability backup involves using an evaluation function at the search leaves that reports the (believed) likelihood of winning from that position. (One way to think of this is as the ratio of winning boards to all boards that agree over the properties used to characterize the board. This number can be estimated by prior experience.) These values are backed up using laws of probability. (We assume statistical independence for disjoint branches for simplicity at this time.) The move is made in favor of best backed up value. One question pursued is whether pathology exists for this methodology as it does for the minimax procedure. (We have recently learned that Dana Nau of Maryland has also been looking at this question.) We are also interested in optimizing this design when search is limited to investigating n nodes ahead. How do we decide where to search?

We have insufficient results to consider a paper at this time.

Perfect information games of chance (Ballard, Reibman)

An extension to the alpha-beta pruning algorithm for game trees with "probability" nodes was discovered. The objective here is to deal with situations (e.g. games) with a chance element added to the usual two-person perfect information zero-sum game setting. The direct concern is to develop a pruning algorithm that trims branches from the search tree (i.e. truncates some search paths) in an optimal way.

A paper has been written presenting an algorithm which, if the search happens to be optimally ordered, reduces the search tree size by an order of magnitude. Empirical studies carried out on non-optimal orderings indicate the algorithm is quite effective in general. Variations of the algorithm have also been empirically studied.

Supported Personnel

Loveland, Donald W. (Principal Investigator)

Biermann, Alan W. (Co-principal Investigator)

Bowyer, Kevin W. (Co-principal Investigator)

Wagner, Robert A. (Co-principal Investigator)

Mutchler, David (Research Assistant)

Reibman, Andrew (Research Assistant)

Truscott, Thomas (Research Associate)

Valtorta, Marco (Research Assistant)

A.M. Thesis

Fabrizio, Robert.

A two-suit bridge player utilizing a combinatorial algorithm.

R. Wagner, supervisor.

(Note: Mr. Fabrizio was not supported by the grant but machine time was provided by the grant. The work studied search strategies in the bridge setting. See Research Status - Wagner)

Publications and Reports

Chronologically ordered

1. Biermann, A., J. Fairfield and T. Beres. Signature tables and learning. To appear in _IEEE Trans. on Systems, Man and Cybernetics_.

2. Biermann, A. Dealing with search. To appear in _Automatic Program Construction Techniques_, (Eds. Biermann, Guiho, Kodratoff), MacMillan Publ. Co.

3. Ballard, B. A search procedure for perfect information games of chance. Submitted to _Artificial Intelligence_. (Note: Prof. Ballard was not funded on this grant but machine time and a research assistant were provided by the grant.)

4. Loveland, D. Performance bounds for binary testing with arbitrary weights. Submitted to _Acta Informatica_.

Presentations, Interactions

Ballard, B. A search procedure for perfect information games of chance. To be presented at AAAI, Pittsburgh, Aug. 1982.

Loveland D. Consultant to Rockland Research Center, Orangeburg, NY (a New York state mental health research center). August 4, 1981. On binary testing and the identification problem.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER **AFOSR-TR- 82-0878**	2. GOVT ACCESSION NO. AD-A120 248	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (*and Subtitle*) SEARCH ALGORITHMS AND THEIR IMPLEMENTATION		5. TYPE OF REPORT & PERIOD COVERED INTERIM, 1 Jul 81-30 Jun 82
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) D.W. Loveland		8. CONTRACT OR GRANT NUMBER(s) AFOSR-81-0221
9. PERFORMING ORGANIZATION NAME AND ADDRESS Computer Science Department Duke University Durham NC 27706		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE61102F; 2304/A2
11. CONTROLLING OFFICE NAME AND ADDRESS Directorate of Mathematical & Information Sciences Air Force Office of Scientific Research Bolling AFB DC 20332		12. REPORT DATE August 1982
		13. NUMBER OF PAGES 17
14. MONITORING AGENCY NAME & ADDRESS(*if different from Controlling Office*)		15. SECURITY CLASS. (*of this report*) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)

Research that has resulted in completed papers involved (1) optimality of search procedures (decision trees) in binary testing; (2) a study of signature table representation for evaluation functions and methods for dynamically improving function accuracy; (3) pruning minimax trees that have been adapted to incorporate moves determined by chance; and (4) the search problem in automated program construction. Preliminary results have been obtained in research on (a) optimizing limited resource use to guide [illegible]se random search; (b) studying search strategies in two-person games [illegible]ormation is (CONTINUED)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

82 10 12 168

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

ITEM #20, CONTINUED: partly concealed; and (c) limiting search in debugging rule sets in one type of expert knowledge system. Other investigations are in progress.

DATE
ILME